Semi-supervised Learning Models
Semi-supervised learning with consistency regularization
- Supervised + consistency regularization (unsupervised)
- 2 forms Consistency regularization: data augmentation and teacher-student

- $\Gamma$-Model, $\Pi$-Model (Ramsmus et al., NIPS 2015; Sajjadi et al., NIPS 2016), simple data augmentation
- Universal data augmentation (UDA) (Xie et al., 2019), complex data-augmentation
- Temporal Ensemble (Laine et al., ICLR 2017), teacher-student
- Mean Teachers (Tarvainen et al., NIPS 2017), teacher-student
- FixMatch (Sohn et al., 2020), simple-complex data-augmentation + teacher-student

Towards NLP
Semi-supervised Learning with Consistency Regularization

- **Supervised part**
  \[
  L(x_l, y_l, \theta) = \text{CE}(q(y_l|x_l), p(y|x_l; \theta))
  \]

- **Consistency regularization part** (a classifier should give consistency output for similar data points (invariant semantics, changing as many pixels of an image without changing its meaning)): 2 forms
  - data augmentation: the two inputs are similar to each other \(x_{ul} \sim x_{ul}^+\),
    \[
    L_c(x_{ul}, \theta) = \mathcal{J}(p(y|x_{ul}; \theta), p(y|x_{ul}^+; \theta))
    \]
  - teacher-student: the student \(p\) tries to match the teacher \(p^+\)'s prediction, teacher and student have similar classifiers \(p \sim p^+\), e.g., same architecture but different parameters
    \[
    L_c(x_{ul}, \theta) = \mathcal{J}(p(y|x_{ul}; \theta), p^+(y|x_{ul}; \theta^+))
    \]

- **Note** there is no clear distinction between the two forms:
  - both \(x_{ul}\) and \(x_{ul}^+\) can be augmentation of the same input
  - classifier \(p\) can also apply data augmentation
  - if the classifier \(p^+\) results in a fixed discrete pseudo-label or continuous distribution (and is not back-propagated) then the method belongs to teacher-student form, else the method is considered data augmentation
\[ L_c(x_{ul}, \theta) = \mathcal{J}(p(y|x_{ul}; \theta), p(y|x_{ul}^+; \theta)) \]

- \(x_{ul}\) and \(x_{ul}^+\) are both augmentation of the same inputs
  - explicit augmentation: for images, invariant transformations such as random crop, flip, rotate, cutout images, changing brightness, color, contrast
  - implicit augmentation: model’s internal stochasticity such as dropout (different passes have different dropouts thus produce different outputs), virtual adversarial examples, mix-up (strange but interesting idea)

- \(\Gamma\)-Model and \(\Pi\)-Model apply simple augmentation: dropout + random crop and flip images

- (Universal data augmentation) UDA applies a complex reinforcement learning strategy to find the best set of augmentations out of 16 augmentation choices (+ their parameters) for each image domain.
Temporal Ensemble and Mean Teacher – teacher-student

\[ L_c(x_{ul}, \theta) = J(p(y|x_{ul}; \theta), p^+(y|x_{ul}; \theta^+)) \]

- **Temporal Ensemble**
  - output \( Z \) of \( p^+ \) of an input is an accumulated prediction of \( p \) of that input
  - \( p^+ = Z = \alpha Z + (1 - \alpha) z \), where \( z \) is the current prediction,
  - \( Z \) is first initialized to be 0

- **Mean Teacher**
  - parameters \( \theta^+ \) of \( p^+ \) is an accumulated parameters of \( p \)
  - \( \theta^+ = \alpha \theta^+ + (1 - \alpha) \theta \), where \( \theta \) is the current parameters
  - \( \theta^+ \) is first initialized to be 0
FixMatch – simple-complex data-augmentation + teacher-student

- **teacher classifier**
  - apply simple data augmentation
  - the classifier’s output is the class with highest probability (larger than some threshold)

- **student classifier**:
  - apply complex data augmentation, i.e., RL strategy
  - student tries to match output of the teacher classifier.
Towards NLP

- Teacher-student seems "easy" to apply
  - Single-/Multi-source cross-lingual NER via Teacher-Student Learning (Wu et al., ACL 2020)

- Data augmentation
  - implicit augmentation: dropout, virtual adversarial examples, ... they work but not as good as explicit augmentation
  - explicit augmentation:
    - it is not easy to do since words are discrete, if we change few words we may change the semantics
    - random noise injection: embeddings noise, spelling error, unigram noising, ...
    - lexical substitutions (wordnet, word-embeddings, masked language model),
    - back translation: translate sentences into different languages then translate them back to original language
    - generative methods
## Current Benchmarks on Text Classifications

<table>
<thead>
<tr>
<th>Datasets</th>
<th>IMDb (25k)</th>
<th>Yelp-2 (560k)</th>
<th>Yelp-5 (650k)</th>
<th>Amazon-2 (3.6m)</th>
<th>Amazon-5 (3m)</th>
<th>DBpedia (560k)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-BERT SOTA</td>
<td>4.32</td>
<td>2.16</td>
<td>29.98</td>
<td>3.32</td>
<td>34.81</td>
<td>0.70</td>
</tr>
<tr>
<td>BERT_LARGE</td>
<td>4.51</td>
<td>1.89</td>
<td>29.32</td>
<td>2.63</td>
<td>34.17</td>
<td>0.64</td>
</tr>
</tbody>
</table>

### Fully supervised baseline

### Semi-supervised setting

<table>
<thead>
<tr>
<th>Initialization</th>
<th>UDA</th>
<th>IMDb (20)</th>
<th>Yelp-2 (20)</th>
<th>Yelp-5 (2.5k)</th>
<th>Amazon-2 (20)</th>
<th>Amazon-5 (2.5k)</th>
<th>DBpedia (140)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>✗</td>
<td>43.27</td>
<td>40.25</td>
<td>50.80</td>
<td>45.39</td>
<td>55.70</td>
<td>41.14</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>25.23</td>
<td>8.33</td>
<td>41.35</td>
<td>16.16</td>
<td>44.19</td>
<td>7.24</td>
</tr>
<tr>
<td>BERT_BASE</td>
<td>✗</td>
<td>18.40</td>
<td>13.60</td>
<td>41.00</td>
<td>26.75</td>
<td>44.09</td>
<td>2.58</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>5.45</td>
<td>2.61</td>
<td>33.80</td>
<td>3.96</td>
<td>38.40</td>
<td>1.33</td>
</tr>
<tr>
<td>BERT_LARGE</td>
<td>✗</td>
<td>11.72</td>
<td>10.55</td>
<td>38.90</td>
<td>15.54</td>
<td>42.30</td>
<td>1.68</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td>4.78</td>
<td>2.50</td>
<td>33.54</td>
<td>3.93</td>
<td>37.80</td>
<td>1.09</td>
</tr>
<tr>
<td>BERT_FINETUNE</td>
<td>✗</td>
<td>6.50</td>
<td>2.94</td>
<td>32.39</td>
<td>12.17</td>
<td>37.32</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>✓</td>
<td><strong>4.20</strong></td>
<td><strong>2.05</strong></td>
<td><strong>32.08</strong></td>
<td><strong>3.50</strong></td>
<td><strong>37.12</strong></td>
<td>-</td>
</tr>
</tbody>
</table>
Thank you!
## Current Benchmarks on Image Classifications

<table>
<thead>
<tr>
<th>Method</th>
<th>CIFAR-10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>40 labels</td>
<td>250 labels</td>
<td>4000 labels</td>
</tr>
<tr>
<td>Π-Model</td>
<td>-</td>
<td>54.26±3.97</td>
<td>14.01±0.38</td>
</tr>
<tr>
<td>Pseudo-Labeling</td>
<td>-</td>
<td>49.78±0.43</td>
<td>16.09±0.28</td>
</tr>
<tr>
<td>Mean Teacher</td>
<td>-</td>
<td>32.32±2.30</td>
<td>9.19±0.19</td>
</tr>
<tr>
<td>MixMatch</td>
<td>47.54±11.50</td>
<td>11.05±0.86</td>
<td>6.42±0.10</td>
</tr>
<tr>
<td>UDA</td>
<td>29.05±5.93</td>
<td>8.82±1.08</td>
<td>4.88±0.18</td>
</tr>
<tr>
<td>ReMixMatch</td>
<td>19.10±9.64</td>
<td>5.44±0.05</td>
<td>4.72±0.13</td>
</tr>
<tr>
<td>FixMatch (RA)</td>
<td>13.81±3.37</td>
<td>5.07±0.65</td>
<td>4.26±0.05</td>
</tr>
<tr>
<td>FixMatch (CTA)</td>
<td>11.39±3.35</td>
<td>5.07±0.33</td>
<td>4.31±0.15</td>
</tr>
</tbody>
</table>